



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/025,572	12/26/2001	Lee Codel Lawson Tarbotton	01.134.01	8371

7590 07/27/2006
Zilka-Kotab, PC
P.O. Box 721120
San Jose, CA 95172-1120

EXAMINER

ZIA, SYED

ART UNIT PAPER NUMBER

2131

DATE MAILED: 07/27/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/025,572

Applicant(s)

TARBOTTON ET AL.

Examiner

Syed Zia

Art Unit

2131

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 15 May 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☐ Claim(s) 1,2,5-10,13-18 and 21-28 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,2,5-10,13-18 and 21-28 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

This office action is in response to amendment, and argument filed on May 15, 2006. Original application contained Claims 1-24. Applicant previously amended Claims 18, 21, 23-24, added new Claims 25-27, and cancelled Claims 3-4, 11-12, and 19-20. Applicant currently amended Claims 1, 19, 17, and added new Claim 28. The amendment and response filed on May 15, 2006 have been entered and made of record. Presently pending claims are 1-2, 5-10, 13-18, 21-28.

Response to Arguments

Applicant's arguments filed on May 15, 2006 have been fully considered but they are not persuasive because of the following reasons:

Regarding Claims, 1, 10, and 11 applicants argued and asserts that the Cozza excerpt relied upon by the Examiner, by virtue of the file details in the cache file, merely indicate whether the file is infected or not. Having the file details in cache file simply does not even suggest that "... said computer file is stored within a clean storage location..." (emphasis added), as claimed. Thus, Cozza fails to meet applicant's claimed language.

This is not found persuasive. Cited prior art, Cozza, teaches a system and method that relates to storing initial state information concerning the file or volume which is being examined for a virus. This information is stored in a cache in a non-volatile storage medium and when

Art Unit: 2131

files are subsequently scanned for viruses, the current state information is compared to the initial state information stored in the cache. If the initial state information differs from the current state information then the file or volume is scanned for viruses which change the state information of the file or volume. If the initial state information and current state information is the same then the file or volume is scanned for a subset of viruses which do not change the state information.

Applicant further argued that Chess, fails to even suggest a scanning technique specifically conditioned on whether "... said computer file is not stored within a clean storage location ..." (emphasis added), as claimed. Chess simply makes no mention of a location of a file being checked, as specifically claimed, and scanning as a function of such location.

This is not found persuasive, Chess clearly teaches a specific scanning method as shown in Fig. 3, Chess teaches a data fork scanning, wherein no scanning is done in two particular cases – the file is known to be non-malicious [Fig. 3 320] or the file contains no or minimal code [Fig. 3 330]. Chess et al. teach a particular scanning method, which provides "rigorous analysis, including ... execution on a simulated environment, ... specifically-instrumented machines..., static analysis and other methods..." [Col. 6 lines 45-55].

Applicant also argued that Ellenberger simply does not even mention using different scanning options for background task scanning versus on-access scanning, as claimed.

This is not found persuasive. Ellenberger teaches a virus scanning method where one or more virus detection algorithms are selected at random for each scan. Specifically, the invention selects "some detection algorithms of the fast group and some of the thorough, but slow executing group" when choosing algorithms [Col. 8 lines 51-534]

Applicant also argued that Hruska clearly fails to meet applicant's claimed technique of background scanning.

This is not found persuasive. Hruska clearly teaches that on-access scanning "intercepts file open and file close operations," [Pg. 129 Col. 2 "On-access virus scanning"] hence making it as a background task.

Applicant also argued that Vignoles clearly fails to disclose any technique that relates to the order in which computer files are scanned, as specifically claimed.

This is not found persuasive Vignoles et al. teach a system for virus scanning that load multiple anti-virus detection drivers, orders them in the likelihood they will detect infection, and scans the files using the drivers in that order [Fig. 1]. One specific reason for this is that if "an early terminate request can be received, for example by a user canceling the scanning process ... the priority ordering carried out at step (6) will have ensured that at least the highest threat viruses will have been scanned for." [Col. 3 lines 44-52].

As a result, cited prior art does implement and teaches method and measures which can be taken to maintain security against malware while reducing the consumed processing resources and its impact upon system responsiveness, as broadly recited in independent and dependent claims.

Applicants clearly have failed to explicitly identify specific claim limitations, which would define a patentable distinction over prior arts. The examiner is not trying to interpret the invention but is merely trying to interpret the claim language in its broadest and reasonable meaning. The examiner will not interpret to read narrowly the claim language to read exactly from the specification, but will interpret the claim language in the broadest reasonable

Art Unit: 2131

interpretation in view of the specification. Therefore, the examiner asserts that the system of cited prior arts does teach or suggest the subject matter broadly recited in independent Claims and in subsequent dependent Claims. Accordingly, rejections for claims 1-2, 5-10, 13-18, 21-28 are respectfully maintained.

Claim Rejections - 35 USC § 112

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claims 1-27 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

3. The Examiner still assert that the term "more thorough" in claims 1, 9, and 17 is a relative term which renders the claim indefinite. The term "more through" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably appraised of the scope of the invention.

Claim 28 recites the limitation "the computer system", and "the user" in line 1-2. There is insufficient antecedent basis for this limitation in the claim.

The term "least substantially" in claim 1, 9, and 17 is a relative term which renders the claim indefinite. The term "least substantially" is not defined by the claim, the specification

Art Unit: 2131

does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 1, 8, 9, 16, 17, 24, and 27-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza [US Patent 5,502,815] in view of Chess et al. [US Patent 6,772,346], Hurska [Virus Detection], and Ellenberger [US Patent 5,684,875].

Cozza teaches a virus scanning method where the file state information is stored in a cache that can be referenced on subsequent scans to increase performance. Specifically, Cozza teaches a computer program product for controlling a computer to scan computer files for malware [Abstract], said computer program product comprising:

- Malware scanning code operable to malware scan all computer files stored within a storage location [Fig. 3 22] as addressed by an operating system [inherent in definition of volume] to identify any computer files stored within said storage location that contain malware [Col. 3 lines 34-36]

- Identification code operable if no computer files containing malware are found in said storage location [Fig. 4 58 “no”] to identify said storage location as a clean storage location [Fig. 4 62]
- When subsequently reading a computer file determination code operable to determine whether or not said computer file is stored within a clean storage location [Fig. 4 40] and:
 - If said computer file is not stored within a clean storage location, then malware scanning said computer file [Fig. 4 40 “no”->42->44]

What is not taught is that if said computer file is stored within a clean storage location, then permitting reading of said computer file without further malware scanning the scanning is done as a background task, or that the malware scanning of clean locations is done using more through scanning options.

Chess et al. teach an antivirus system where files are sent to a central location for scanning in a distributed system. A specific scanning method is also taught in Fig. 3, wherein no scanning is done in two particular cases – the file is known to be non-malicious [Fig. 3 320] or the file contains no or minimal code [Fig. 3 330]. In addition, Cozza et al. teach that “at the present time there are no such viruses that affect the resource forks of files on Apple Macintosh computer without changing the resource forks of files ... without changing the resource fork length, so no scanning would be necessary in step (50)” [Col. 4 lines 41-46]. It is also well known in the art that the resource fork does not exist on 68K Mac executables.

It would have been obvious to one of ordinary skill in the art, that if the system the invention of Cozza et al. was running on was a 68K Mac and the file was known to be non-

Art Unit: 2131

malicious or contained minimal code and was in the data cache, to not scan the file. Cozza et al. teach that the subset of viruses, which do not effect cached data, should be scanned. Cozza et al. additional teach that in 68K Macs this scan for resource forks is unnecessary. As for data fork scanning, Cozza et al. remain silent on the specifics of what a scan entails. Chess et al. teach a particular scanning method, which provides “rigorous analysis, including ... execution on a simulated environment, ... specifically-instrumented machines..., static analysis and other methods...” [Col. 6 lines 45-55]. One of skill in the art would wish to use a scanner that uses rigorous analysis, and thus it would be obvious to use this scanner as the scanner in the system of Cozza et al, which in some cases does not scan.

The combination of Cozza et al. and Chess et al. still does not teach malware scanning the scanning is done as a background task, or that the malware scanning of clean locations is done using more through scanning options.

Hruska gives an overview of virus detection methods. In regards to On-Access scanning it is taught that it is “much safer then any alternative” [Pg. 129 Col. 2 “On-access virus scanning”]. It is also taught that on-access scanning “intercepts file open and file close operations,” [Pg. 129 Col. 2 “On-access virus scanning”] hence it is a background task.

It would have been obvious to one of ordinary skill in the art to apply the methods of Cozza et al. to an on-access scanner. On-access scanners were well known in the art, and Hruska teaches that it is much safer then any alternative for virus interception. This scanning would then be performed as a background task.

What is still not taught is that the malware scanning of all computer files stored within a storage location as a background task that occurs as a function of when an associated computer

Art Unit: 2131

system is at least substantially idle and is performed with more thorough scanning options selected than for on-access scanning applied to computer files not stored within clean storage locations and being accessed by a user.

Ellenberger teaches a virus scanning method where one or more virus detection algorithms are selected at random for each scan. Specifically, the invention selects “some detection algorithms of the fast group and some of the thorough, but slow executing group” when choosing algorithms.

It would have been obvious to use the methods of Ellenberger to select the specific virus detection algorithms used when scanning as the algorithm used by the combined invention of Cozza et al. Ellenberger teaches that using all algorithms would be too slow, and the method taught “surpasses current state of the art without undue consumption of resources” [Col. 8 lines 51-54]. Since some scanners will inherently be more or less thorough than others depending on the criteria chosen to measure thoroughness [See 112 rejection – ‘more thorough’ being a relative term] in some cases the scanning done for a referenced secure location will have more thorough scanning options than a default location.

6. In regards to claim 9, what is claimed is a method that corresponds to the actions of the computer program product claimed in claim 1. The same rationale of rejection applies.

7. In regards to claim 17, what is claimed is an apparatus for scanning computer files for malware, the apparatus containing logic operable to perform the same operations as the computer program product of claim 1. The invention of Cozza et al. is specifically identified as a “method and apparatus for detecting the existence of a computer virus on a computer.” [Col. 1 lines 6-7]

Art Unit: 2131

8. In regards to claims 8, 16, and 24 the invention of Cozza et al. is specifically drawn to anti-virus scanning.

9. In regards to claim 27, it is inherent that said storage location “share a common logical storage location as views by the operating system” by the definition of volume. As for “said logical storage location includes computer files sharing similar characteristics,” Cozza et al. is drawn to the Mac Classic file system [Col. 5 lines 8-20] wherein each file consist of a data and resource fork [Col. 4 lines 24-28]

10. In regards to claim 28, as claimed in claim 1, wherein the background task avoids interference with a responsiveness of the computer system when the user starts to use the computer system Ellenberger [Col. 8 lines 51-54].

11. Claims 2, 10, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza et al. as applied to claims 1, 9, and 17 above, and further in view of Colorado et al. [How to Exclude Folders from NAV Virus Scanning?].

Cozza et al. teach all the limitations of claims 1, 9, and 17. What Cozza et al. do not teach is that said malware scanning of all computer files stored within a storage location is performed upon a set of user specified storage locations from within all storage locations accessible to a user.

Colorado et al. teach that in Norton Antivirus that it is possible to exclude folders from scanning for on-access and on-demand scanning [Reply by John Robson, Wed Apr 4 2001 3:40pm] and that it is desirable to do so in some cases for performance reasons [Reply by Colorado Dave, Wed Apr 4 2001 6:35pm].

It would have been obvious to one of ordinary skill in the art to modify Cozza et al. to exclude certain folders from scanning. Colorado et al. teach that doing so would be beneficial in certain cases since certain high-activity and low-risk folders would unnecessarily increase the system load due to constant on-access scanning. In regards to claim language, the “user specified storage locations from within all storage locations” would be the set of all storage locations, minus the specified excluded storage locations.

12. Claims 5, 13, and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza et al. as applied to claims 1, 9, and 17 above, and further in view of Symantec.com [Norton Antivirus 2001 for Windows 2000/NT/Me/98/95].

Cozza et al. teach all the limitations of claims 1, 9, and 17 above. What Cozza et al. do not explicitly teach is that a compute file is malware scanned before being written to a clean storage location.

Symantec.com teaches that their virus scanner “scans files you download from the web, as well as attachments you get through email.” [Paragraph 1, lines 5-7]

It would have been obvious to one of ordinary skill in the art to scan files being written to any storage location. This was common practice in the art at the time in order to attempt to catch certain classes of viruses at the point of infection, and Symantec.com specifically teaches this feature.

Art Unit: 2131

13. Claims 6, 14, 22, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza et al. as applied to claims 1, 9, and 17 above, and further in view of Polk et al. [A Guide to the Selection of Anti-Virus Tools and Techniques].

14. In regards to claims 6, 14, and 22 Cozza et al. teach all the limitations of claim 1, 9, and 17 above. In addition, Cozza et al. teach that verification includes validating that the cache's version number is not out of date [Col. 3 lines 57-58]. Cozza et al. also teach that the malware scanning code uses malware definition data to identify malware [Col. 1 line 66 – Col. 2 line 6]

What Cozza et al. do not teach is that the malware data is updated, and that upon updating the malware definition data the clean storage location is no longer identified as a clean storage area until it has been malware scanned using said updated malware definition data and no computer files containing malware are found in said storage location.

Polk et al. teach that in regards to signature scanners, "New viruses are discovered every week. As a result, virus scanners are immediately out of date [and] ... procedures must be devised for distribution of updates" [Pg. 13 "Administrative Overhead"]

It would have been obvious to one of ordinary skill in the art to allow for the update of the virus scanner used in the invention of Cozza et al. Polk et al. teaches that the scanners must be kept up to date to be effective, which means that such functionality is crucial to the scanner. As a result of an update, the version number as referenced in Cozza et al. would change, which would invalidate the cache i.e. the cached data is no longer treated as referencing clean storage locations [Fig. 3 28 "no"]. Upon rescanning, the cache would be updated and clean storage locations identified in the absence of viruses [Fig. 4 58->62]

Art Unit: 2131

15. In regards to claim 26, Polk et al. additionally teaches that scanners “may be employed reactively ... scanning the system at regular intervals” [Pg. 11 4.1 Paragraph 2]. Combined with the fact that scanners go immediately out of date, it would be obvious for one of ordinary skill in the art to rescan upon an update to definition. Since scanning is done as a background task, this would also entail a background task.

16. Claims 7, 15, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza et al. as applied to claims 6, 14, and 22 above, and further in view of Davidson et al. [Unix Anti-Virus Software] and Vignoles et al. [US Patent 6,938,161].

Cozza et al. teach all the limitations of claims 6, 14, and 22. What they do not teach is that “when the storage area is being malware scanned with said updated malware definition data, computer files written to said storage location after said storage location after said storage location was previously identified as a clean storage location are malware scanned before computer files that are unaltered since said storage location was previously identified as a clean storage location.”

Davidson et al. teach that tripwire, a modification detection tool, should be run “as often as you like, and on the files you feel are prone to infection.” [Reply 4 date July 23 1997]

Vignoles et al. teach a system for virus scanning that load multiple anti-virus detection drivers, orders them in the likelihood they will detect infection, and scans the files using the drivers in that order [Fig. 1]. One specific reason for this is that if “an early terminate request can be received, for example by a user canceling the scanning process ... the priority ordering

Art Unit: 2131

carried out at step (6) will have ensured that at least the highest threat viruses will have been scanned for.” [Col. 3 lines 44-52]

It would have been obvious to one of ordinary skill in the art, that upon updating of the malware definition data, to scan files within the clean storage location that are altered before files that are unaltered. The techniques of Cozza et al. have a resemblance to modification detection tools, in that if a file is modified it pays special attention to the file. Vignolet et al. teach that since there is a possibility that a scan might be cancelled, it is important to insure that the highest threats will have been scanned for first. Since a file that has been modified has a higher threat to contain a virus than a file that is unmodified, it would be obvious to one in the art to scan that file first in case a complete scan would be terminated.

1. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cozza et al. as applied to claim 1 above, and further in view of Faltstrom et al. (RFC 1740: MIME Encapsulation of Macintosh Files – MacMIME).

Cozza et al. teach all the limitations of claims 1. What Cozza et al. do not explicitly teach is that if the file is stored within said clean storage location, then said computer file is permitted to be read without further time spent on malware-related processing.

Faltstrom et al. teach in relation to the sending of Macintosh files that “Documents which lack a data fork must be sent as AppleSingle” [Pg. 3 2c Paragraph 2]

It would have been obvious to one of ordinary skill in the art, if files were being scanned that do not include a data fork, that no further malware scanning would take place. As referenced in claim 1 Cozza et al. already teach that given a clean storage location resource forks

Art Unit: 2131

are not scanned. If a data fork is not present, then the data fork scanning cannot proceed and thus no further malware scanning is done.

Conclusion

\Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed Zia whose telephone number is 571-272-3798. The examiner can normally be reached on 9:00 to 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on 571-272-3795. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2131

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

sz

July 12, 2006

A handwritten signature in black ink, appearing to be 'S. Ma' with a stylized flourish at the end.